

JavaScript 语法手册及编程练习

作者：王琦

创建日期：2017 年 5 月 3 日

最后修改：2017 年 5 月 4 日

摘要：本文提供了 ES6 标准下的部分 JavaScript 语法，每个知识点都由范例、参考链接和练习组成，读者可在阅读内容后完成相应练习以巩固理解。

1. 数组的遍历

1.1 语法

第一种遍历方式 — 通过下标访问数组，并进行数组求和：

```
let acc = 0; // accumulate, 累和
const a = [10, 20, 30]; // 定义数组 a
for (let i = 0; i < a.length; ++i)
  acc += a[i];
console.log(acc); // 在控制台打印累和的值（应为 60）
```

第二种遍历方式 [for...of](<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/for...of>):

```
let acc = 0;
const a = [10, 20, 30];
for (const val of a)
  acc += val;
```

第三种遍历方式 [for...in](<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/for...in>):

```
let acc = 0;
const a = [10, 20, 30];
for (const i in a)
  acc += a[i];
```

1.2 练习

请完成下述练习：

- [LeetCode 1. Two Sum](<https://leetcode.com/problems/two-sum/#/description>)

2. 对象

2.1 语法

2.2 练习

链表 (List)

请完成下述练习：

- [LeetCode 2. Add Two Numbers](https://leetcode.com/problems/add-two-numbers/)

树 (Tree)

用递归地方法完成下述两道练习：

- [LeetCode 104. Maximum Depth of Binary Tree](https://leetcode.com/problems/maximum-depth-of-binary-tree/#/description)
- [LeetCode 94. Binary Tree Inorder Traversal](https://leetcode.com/problems/binary-tree-inorder-traversal/#/description)

Set/Map 容器

1.1 语法

[MDN - Set](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/set)

[MDN - Map](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/map)

1.2 时间复杂度

根据 ES6 标准和 [stackoverflow 上的问答](http://stackoverflow.com/questions/31091772/javascript-es6-computational-time-complexity-of-collections), ES6 规定了 Set/Map 实现（比如说 Set.prototype.has）的复杂度应该是 sub-linear （亦即 $O(n)$ ）的。

根据 [es6 Map and Set complexity, v8 implementation](http://stackoverflow.com/questions/33611509/es6-map-and-set-complexity-v8-implementation), V8 引擎可以在 $O(1)$ 进行容器访问。

1.3 练习

使用 Map 完成 [LeetCode 1. Two Sum](https://leetcode.com/problems/two-sum/#/description), 并和`双重 for 循环`的方法比较, 观察这个实现的耗时是否减少了。